PATENT ABSTRACTS OF JAPAN

(11)Publication number : 07-334468
(43)Date of publication of application : 22.12.1995

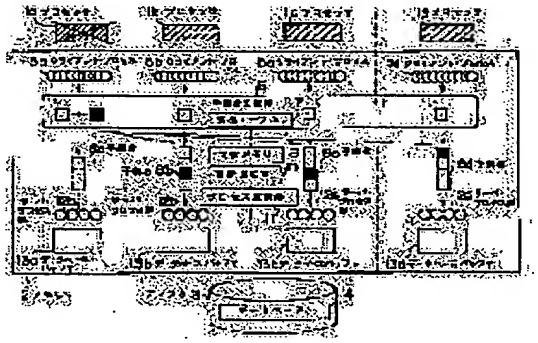
(51)Int.Cl. G06F 15/16
G06F 15/16

(21)Application number : 06-125484
(22)Date of filing : 07.06.1994

(71)Applicant : TOSHIBA CORP
(72)Inventor : KATO NORIHIRO

(54) LOAD DISTRIBUTION SYSTEM

(57)Abstract:
PURPOSE: To provide a load distribution system capable of efficiently distributing load when the rearrangement of processes among respective processors due to a processor fault or the like is executed.
CONSTITUTION: This load distribution system is provided with a process monitoring part 11 for monitoring which processor client processes 5a to 5d and server process groups 12a to 12d are arranged under the control of, a procedure distributing part 6 for distributing a procedure requested from the client processes 5a to 5d to prescribed server process groups 12a to 12d and a control means for controlling the distributing part 6 so that procedures requested from plural client processes 5a, 5b are distributed to one server process group 12b at the time of detecting a processor 1b in which plural client processes 5a, 5b and plural server process groups 12a, 12b are arranged due to the generation of a fault in the other processor 1a.



LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

(19)日本国特許庁（ＪＰ）　(12)公開特許公報（Ａ）　(11)特許出願公開番号

特開平7－334468

(43)公開日　平成７年(1995)12月22日

(51)Int.Cl.[6]	識別記号	庁内整理番号	ＦＩ	技術表示箇所
Ｇ０６Ｆ　15/16	３８０　Ｚ			
	４７０　Ｓ			

審査請求　未請求　請求項の数２　ＯＬ　（全７頁）

(21)出願番号　特願平6－125484

(22)出願日　平成６年(1994)６月７日

(71)出願人　000003078
株式会社東芝
神奈川県川崎市幸区堀川町72番地

(72)発明者　加藤　宣弘
東京都青梅市末広町２丁目９番地　株式会社東芝青梅工場内

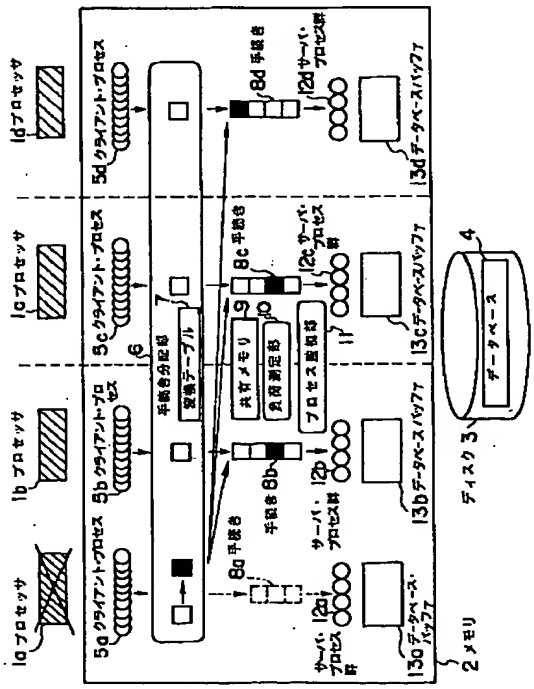
(74)代理人　弁理士　鈴江　武彦

(54)【発明の名称】　負荷分散方式

(57)【要約】

【目的】プロセッサ障害等によるプロセッサ間でのプロセス再配置が行われた際に、効率的に負荷分散を行うことを可能とする負荷分散方式を提供する。

【構成】クライアント・プロセス５ａ～５ｄ及びサーバ・プロセス群１２ａ～１２ｄがいずれのプロセッサ下に配置されているのかを監視するプロセス監視部１１と、上記クライアント・プロセス５ａ～５ｄから要求された手続きを所定のサーバ・プロセス群１２ａ～１２ｄに分配する手続き分配部６と、他のプロセッサ１ａの障害により複数のクライアント・プロセス５ａ～５ｂ及びサーバ・プロセス群１２ａ～１２ｂが配置されているプロセッサ１ｂを検出した場合に、この複数のクライアント・プロセス５ａ～５ｂから要求された手続きが一つのサーバ・プロセス群１２ｂに分配されるように上記手続き分配部６を制御する制御手段とを具備してなることを特徴とする。



1

【 特許請求の範囲】

【 請求項1 】　複数のプロセッサ各々の制御下に、クライアント・プロセスと、このクライアント・プロセスから要求された手続きを所定のメモリ領域を共有しながら実行する複数のプロセスからなるサーバ・プロセス群とを配置し、いずれかのプロセッサに障害が発生したときに、そのプロセッサ下に配置されていた上記クライアント・プロセス及びサーバ・プロセス群を他の特定のプロセッサ下に再配置させることにより処理を継続するマルチプロセッサ計算機において、

上記クライアント・プロセス及びサーバ・プロセス群がいずれのプロセッサ下に配置されているのかを監視するプロセス監視手段と、上記クライアント・プロセスから要求された手続きを所定のサーバ・プロセス群に分配する手続き分配手段と、上記プロセス監視手段の監視結果により複数のクライアント・プロセス及びサーバ・プロセス群が配置されているプロセッサを検出した場合に、この複数のクライアント・プロセスから要求された手続きが一つのサーバ・プロセス群に分配されるように上記手続き分配手段を制御する制御手段とを具備してなることを特徴とする負荷分散方式。

【 請求項2 】　各プロセッサの負荷を測定する負荷測定手段と、この負荷測定手段の測定結果によりプロセッサ間に負荷の偏りがあると判断した場合に、負荷の高いプロセッサ下のサーバ・プロセス群に本来分配されるべき手続きの一部を負荷の低い他のプロセッサ下のサーバ・プロセス群へ分配されるように上記手続き分配手段を制御する制御手段とを具備してなることを特徴とする請求項1 記載の負荷分散方式。

【 発明の詳細な説明】

【０００１】

【 産業上の利用分野】本発明は、フォールト・トレンランスを実現するマルチプロセッサ計算機に適用して好適な負荷分散方式に係り、特にプロセッサ障害等によるプロセッサ間でのプロセス再配置が行われた際に、効率的な負荷分散を行うことを可能とする負荷分散方式に関する。

【０００２】

【 従来の技術】従来のフォールト・トレラント計算機の中には、あるプロセッサが障害を起こした場合に、そのプロセッサ下で実行していたプロセスを他のプロセッサ下で継続して実行することのできる計算機がある。そのような特徴を持つフォールト・トレラント計算機として、例えば、文献「 P.A.Bernstein,"Sequoia:A Fault-Tolerant Tightly Coupled Multiprocessor for Transaction Processing",IEEE COMPUTER,Vol.21,No.2,Feb.1988" 」で述べられているようなノンライト・スルー・キャッシュを用いたマルチプロセッサ計算機がある。このような計算機においてオンライン・トランザクション処理を行う場合には、データベースにアクセスするサーバ・プロセスを特定のプロセッサで実行するようなプロセス構成を取る。それは以下のような理由による。

2

【０００３】ノンライト・スルー・キャッシュを採用するマルチプロセッサ計算機の場合には、キャッシュ内での共有データの書き換えがすぐにメモリへ反映されないために、他のプロセッサによる共有データへのアクセスは、それがメモリへ書き戻されるのを待たなければならない。この待ち時間はプロセッサの使用効率を低下させることとなる。しかし、同じプロセッサ内で動作する他のプロセスが共有データにアクセスする場合には、キャッシュ・ヒットするために、メモリへの書き戻しを待つ必要がない。従って、データベースにアクセスするサーバ・プロセスのように、共有データにアクセスする複数のプロセスはできる限り同じプロセッサで実行するのが望ましいと言える。

【０００４】一方、オンライン・トランザクション処理を実行する場合には、システムのスループットを最大限引き出すために、プロセスを複数のプロセッサを並列に動作させる必要がある。

【０００５】このような二つの相反する要件を満足するためには、グループ間でのデータの共有度ができる限り低くなるようにプロセスをグループ化し、かつ、それぞれのグループを特定のプロセッサに割り付けるようにする。つまり、同じグループに属するプロセスは同じプロセッサ下で動作するので待ち時間が少なくなり、別のグループに属するプロセス相互間ではデータの共有度が低いので待ち状態になる場合が低くなるため、プロセッサを効率的に使用することができる。

【０００６】従来のオンライン・トランザクション処理システムは、トランザクション・モニタと呼ばれるソフトウェアを用いて、トランザクションをクライアント・サーバのプロセス構成で処理する。図6 にトランザクション・モニタを用いたプロセス構成を示す。

【０００７】クライアント・プロセス５ａ ～５ｄが実行してほしい手続き８ａ ～８ｄをコールすると、トランザクション・モニタがその手続き８ａ ～８ｄを実行可能なサーバ・プロセス群１２ａ ～１２ｄのキューに引き渡す。サーバ・プロセス群１２ａ ～１２ｄはキューから要求を取り出して指定された手続き８ａ ～８ｄを実行した後、クライアント・プロセス５ａ ～５ｄへ応答する。

【０００８】このようなトランザクション・モニタを上述のフォールト・トレラント計算機で動作させる場合、以下の3 つの方針に基づいて構成することにより、高性能なシステムを構築できる。

【０００９】まず第一に、データベース・バッファを共有するサーバ・プロセスを同じプロセッサに割り付ける。この理由は上述の通りである。第二に、一つのプロセッサに割り付けられた全てのサーバ・プロセスに対して、一つのデータベース・バッファを共有させる。なぜならば、異なるデータベース・バッファで同じデータを

3

共有する場合、データの一貫性を取るための制御が必要であり、その制御はかなりオーバヘッドの高い処理となるからである。第三に、システムのスループットを最大限に引き出すために、すべてのプロセッサにおいてほぼ均一な負荷でサーバ・プロセスを動作させる。以上の方針に基づいたフォールト・トレラント計算機におけるトランザクション・モニタの構成例を図6に示している。なお、ここでは、一つのデータベース・バッファを共有する複数のサーバ・プロセスをサーバ・プロセス群と呼ぶこととする。

【００１０】

【発明が解決しようとする課題】図6に示すような構成で、あるプロセッサに障害が起こった場合、フォールト・トレラント計算機のリカバリ機能が働き、障害プロセッサで実行していたサーバ・プロセスを自動的に他の特定のプロセッサで実行させることになる。

【００１１】この結果、一つのプロセッサで複数のサーバ・プロセス群を動作させることになり、また、障害プロセッサの処理を引き継いだプロセッサの負荷が高くなるためにプロセッサ間での負荷が不均等になる。そのために上述の方針から外れた構成でトランザクション・モニタを運用することになり、性能低下を招くことになる。

【００１２】いま、図6に示すプロセッサ１ａに障害が発生したとすると、図7に示すようにクライアントプロセス５ａ及びサーバ・プロセス群１２ａは、プロセッサ１ｂ下で動作することとなる。即ち、プロセッサ１ｂでは2つのサーバ・プロセス群１２ａ～１２ｂが動作することになる。従って、２つのデータベース・バッファ１３ａ～１３ｂに重複してデータを持つ可能性があり、データの一貫性を保証するための制御にかかるオーバヘッドが高くなる。また、プロセッサ１ｂでは2つのサーバ・プロセス群１２ａ～１２ｄを実行しなければならないので、プロセッサ１ｂが過負荷状態になる可能性がある。プロセッサ１ｃあるいは１ｄは、適切な負荷状態出運用されているので、プロセッサ１ｂだけが過負荷になれば、負荷の不均等な状態が生じることになる。

【００１３】本発明は、上記実情に鑑みてなされたものであり、プロセッサ障害等によるプロセッサ間でのプロセス再配置が行われた際に、効率的な負荷分散を行うことを可能とする負荷分散方式を提供することを目的とする。

【００１４】

【課題を解決するための手段】本発明は、複数のプロセッサ各々の制御下に、クライアント・プロセスと、このクライアント・プロセスから要求された手続きを所定のメモリ領域を共有しながら実行する複数のプロセスからなるサーバ・プロセス群とを配置し、いずれかのプロセッサに障害が発生したときに、そのプロセッサ下に配置されていた上記クライアント・プロセス及びサーバ・プロセス群を他の特定のプロセッサ下に再配置させることにより処理を継続するマルチプロセッサ計算機において、上記クライアント・プロセス及びサーバ・プロセス群がいずれのプロセッサ下に配置されているのかを監視するプロセス監視手段と、上記クライアント・プロセスから要求された手続きを所定のサーバ・プロセス群に分配する手続き分配手段と、上記プロセス監視手段の監視結果により複数のクライアント・プロセス及びサーバ・プロセス群が配置されているプロセッサを検出した場合に、この複数のクライアント・プロセスから要求された手続きが一つのサーバ・プロセス群に分配されるように上記手続き分配手段を制御する制御手段とを具備してなることを特徴とする。

【００１５】また、本発明は、各プロセッサの負荷を測定する負荷測定手段と、この負荷測定手段の測定結果によりプロセッサ間に負荷の偏りがあると判断した場合に、負荷の高いプロセッサ下のサーバ・プロセス群に本来分配されるべき手続きの一部を負荷の低い他のプロセッサ下のサーバ・プロセス群へ分配されるように上記手続き分配手段を制御する制御手段とを具備してなることを特徴とする。

【００１６】

【作用】本発明によれば、図7に示したように一つのプロセッサで複数のサーバ・プロセス群１２ａ～１２ｂを動作させている場合に、クライアント・プロセス５ａ～５ｂからの手続きを一つのサーバ・プロセス群、例えばサーバ・プロセス群１２ｂのキューのみへ引き渡すように分配制御する。これにより、一つのプロセッサ下では一つのサーバ・プロセス群だけが動作することになり、即ち、そのデータベース・バッファだけが使用されることになる。つまり、もう一方のデータベース・バッファは使用されていない状態なので、二つのバッファ間でのデータの一貫性を保証する必要はなくなり、その処理に必要なオーバヘッドをなくすことができる。

【００１７】また、障害を発生させたプロセッサ１ａ下に配置されていたサーバ・プロセス群１２ａの実行を引き継いだプロセッサ１ｂには、それまでに実行していたサーバ・プロセス群１２ｂの負荷に加えて、新たに実行するサーバ・プロセス群１２ａの負荷がかかる。したがって、上述のようにプロセッサ１ｂ下で動作するサーバ・プロセス群をサーバ・プロセス群１２ｂの一つだけに限定したとしても、プロセッサ１ｂが過負荷状態になることがある。

【００１８】本発明によれば、図7においてプロセッサ１ｂが過負荷状態になった場合に、クライアント・プロセス５ａから要求された手続きの一部をプロセッサ１ｃ～１ｄのサーバ・プロセス群１２ｃ～１２ｄのキューへ割り振るようにする。これによりプロセッサ１ｂの負荷を軽減させると共に、プロセッサ１ｃ～１ｄの負荷を増大させることになり、システム全体として負荷の均等化

4

5

を実現することができる。

【００１９】

【実施例】以下図面を参照して本発明の一実施例を説明する。図1は同実施例に係るシステムの概略構成を示す図である。同システムでは、それぞれのサーバ・プロセス群１２ａ～１２ｄが実行する手続きについて、他のすべてのサーバ・プロセス群１２ａ～１２ｄが相互に実行できるようにするため、その手続きを内容は同じで名前が異なるような構成として登録しておく。ここでは、説明を簡単にするために、図1における手続き８ａ～８ｄを名前は異なるが内容が等しい手続きであるとものとする。

【００２０】図1において、負荷測定部１０は、定期的に、あるいは手続き分配部6からの指示により各プロセッサ１ａ～１ｄの負荷を測定し、その結果を共有メモリ９へ書き込む。プロセッサ１ａ～１ｄの負荷としては、プロセッサ使用率やレディ・キューの長さ等を用いることが考えられる。ここではプロセッサ１ａ～１ｄの使用率をオペレーティング・システムが提供するコマンドにより測定するものとする。

【００２１】プロセス監視部１１は、サーバ・プロセス群１２ａ～１２ｄが配置されているプロセッサがどれであるかを監視しており、一つのプロセッサで複数のサーバ・プロセス群を実行していることを見つけた場合、手続き分配部6にその旨を伝える。また、プロセス監視部１１は、プロセッサ１ａ～１ｄの障害等によりプロセッサ間でのプロセスの再配置が行われたときに起動される。

【００２２】手続き分配部6は、変換テーブル7を保持し、この変換テーブル7に記述された内容にしたがって手続きを変換し、その手続きを適切なサーバ・プロセス群のキューへ割り振る。また、手続き分配部6は、プロセス監視部１１からの指示を受けて起動され、変換テーブル7を変更する。そして、変換テーブル7を変更した後に、負荷測定部１０に各プロセッサ１ａ～１ｄの負荷を測定させる。その結果、プロセッサ１ａ～１ｄの負荷に偏りがある場合には、変換テーブル7を変更する。

【００２３】なお、ここでは、手続き分配部6をクライアント・プロセス５ａ～５ｄとは別のプロセスとしているが、クライアント・プロセス５ａ～５ｄで実行されるプログラムの一部に手続き分配部6の動作を含めることも可能である。

【００２４】図1において、プロセッサが全て正常な状態のときは、サーバ・プロセス群１２ａ～１２ｄはそれぞれプロセッサ１ａ～１ｄに割り付けられて、クライアント・プロセス５ａ～５ｄから要求された手続き８ａ～８ｄをそれぞれ実行しているものとする。このときの手続き分配部6の変換テーブル7は、図2に示す通りである。また、負荷測定部１０は、各プロセッサ１ａ～１ｄの使用率を定期的に測定して、その結果を共有メモリ９

6

へ記録する。この値は、負荷の再分散を行うときに用いる。ここでは、各プロセッサ１ａ～１ｄの使用率はすべて６０％程度であるとする。

【００２５】いま、プロセッサ１ａに障害が発生し、フォールト・トレラント計算機のリカバリ機能によりクライアント・プロセス５ａとサーバ・プロセス群１２ａとがプロセッサ１ｂに再配置されたとする。プロセス監視部１１は、サーバ・プロセス群１２ａ～１２ｄが配置されているプロセッサを調べ、プロセッサ１ｂ下でサーバ・プロセス群１２ａ～１２ｂの二つのサーバ・プロセス群が配置されていることを検出して、手続き分配部6へプロセッサ１ｂ下で二つのサーバ・プロセス群が実行されている旨を通知する。

【００２６】プロセス監視部１１からの通知を受けた手続き分配部6は、変換テーブル7を図3に示すように変更し、クライアント・プロセス５ａから要求された手続き８ａをすべて手続き８ｂに変換してサーバ・プロセス群１２ｂのキューへ割り当てるようにする。これにより、サーバ・プロセス群１２ａのデータベース・バッファ１３ａは使われなくなり、データベース・バッファ１３ａ～１３ｂ間でのデータの一貫性を保証する処理が必要なくなる。

【００２７】ここで、プロセッサ１ｂ～１ｄの負荷がほぼ均一であれば問題ない。しかし、通常は全てのプロセッサが正常に動作している場合を想定して負荷分散がなされているので、プロセッサ１ａの負荷を引き継いだプロセッサ１ｂの負荷が、他のプロセッサ１ｃ～１ｄの負荷よりも高くなる可能性が高いと考えられる。

【００２８】そこで、手続き分配部6は、負荷測定部１０へ各プロセッサ１ｂ～１ｄの使用率を測定することを指示する。負荷測定部１０の測定の結果、プロセッサ１ｂの負荷が他のプロセッサ１ｃ～１ｄに比べて顕著に大きいことが判明した場合に、手続きの分配部6が負荷の再分散を行う。例えば、プロセッサ１ｂの使用率が１００％で、それ以外のプロセッサ１ｃ～１ｄの使用率が６０％程度であったとすると、プロセッサ１ｂの負荷をプロセッサ１ｃ～１ｄへ分散させるように、手続き分配部６の変換テーブル7を変更する。例えば、正常時にプロセッサ１ｂ～１ｄの使用率がほぼ等しかったとすると、プロセッサ１ａにかかっていた負荷を三つのプロセッサ１ｂ～１ｄに均等に分散すればよいことになる。そこで、手続き分配部6は、変換テーブル7を図4に示すように変更する。

【００２９】ここで、プロセッサに障害が起こった場合の動作を説明する。図5は、同実施例の動作を説明するためのフローチャートである。まず、各プロセッサ１ａ～１ｄが実行しているサーバ・プロセス群の数を調べる（図5のステップＡ１）。サーバ・プロセス群１２ａ～１２ｄを特定のプロセッサ１ａ～１ｄで実行するために、オペレーティング・システムに何らかの機構が用意

7

されているはずである。そして、この機構を用いることによりサーバ・プロセス群１２ａ～１２ｄがどのプロセッサで実行されているかを知ることができる。

【００３０】次に、複数のサーバ・プロセス群を実行しているプロセッサがあるかどうかを判定する（図5のステップＡ２）。通常、すべてのプロセッサ１ａ～１ｄでサーバ・プロセス群１２ａ～１２ｄのいずれかが実行されているので、あるプロセッサが故障して、そのプロセッサ下に配置されているサーバ・プロセス群が他のプロセッサへ再配置された場合には、そのプロセッサでは複数のサーバ・プロセス群を実行していることになる。このため、プロセッサ障害発生時には、ほとんどの場合この条件に該当することとなる（図5のステップＡ２のＹ）。

【００３１】複数のサーバ・プロセス群が実行されているプロセッサが存在する場合（図5のステップＡ２のＹ）、これまでそれぞれのサーバ・プロセス群へ割り当てていた手続きを一つのサーバ・プロセス群へ割り当てるように、手続き分配部6の変換テーブル7を変更する（図5のステップＡ３）。

【００３２】次に、各プロセッサ１ａ～１ｄの使用率を調べ（図5のステップＡ４）、プロセッサの使用率に偏りがないか調べる（図5のステップＡ５）。故障プロセッサに対する負荷を引き継いだプロセッサの使用率は、他のプロセッサ使用率に比べて高くなるので、ほとんどの場合は該当することとなる（図5のステップＡ５のＹ）。

【００３３】ここで、故障が起こる前の各プロセッサ１ａ～１ｄの使用率に基づいて、故障プロセッサの負荷をどのように他のプロセッサへ分配するかを決定する（図５のステップＡ６）。なお、本発明は、ここに示した実施例に限定されるものではなく、その請求範囲を逸脱しない範囲で適宜変更可能である。

【００３４】

【発明の効果】上述したように本発明によれば、フォールト・トレランスを実現するプロセッサ計算機においてトランザクション処理を行う場合に、プロセッサが障害を発生させたためにそのプロセッサで実行されていたサーバ・プロセス群を他のプロセッサに引き継がれた時の負荷分散を効率よく実現できる。

8

【００３５】サーバ・プロセス群の実行を引き継いだプロセッサ下で複数のサーバ・プロセス群を実行すると、複数のデータベース・バッファ間のデータの一貫性を保証するための処理が必要となり、そのプロセッサの負荷が高くなる。

【００３６】本発明では、複数のクライアント・プロセスからの手続きを一つのサーバ・プロセス群のキューのみに割り振ることにより、この問題を解決している。また、サーバ・プロセス群の実行を引き継いだプロセッサには、故障プロセッサにかかっていた負荷が追加されるため、他のプロセッサに比べて負荷が高くなる。

【００３７】本発明では、負荷の高いプロセッサのサーバ・プロセス群のキューへ割り振られていた手続きを他のプロセッサのサーバ・プロセス群のキューへ割り振られるように変更することにより、負荷の均等化を図ることができる。

【図面の簡単な説明】

【図1】同実施例に係るシステムの概略構成を示す図。

【図2】プロセッサが正常な時の変換テーブルの例を示す図。

【図3】障害プロセッサで実行していたサーバ・プロセス群を別のプロセッサで動作させた後の変換テーブルの例を示す図。

【図4】負荷を分散した後の変換テーブルの例を示す図。

【図5】同実施例の動作を説明するためのフローチャート。

【図6】従来のフォールト・トレラント計算機でトランザクション・モニタを動作させた時のプロセス構成を示す図。

【図7】従来のフォールト・トレラント計算機でプロセッサに障害が起こったときのプロセスの配置を示す図。

【符号の説明】

１ａ～１ｄ…プロセッサ、２…メモリ、３…ディスク、４データベース、５ａ～５ｄ…クライアント・プロセス、６…手続き分配部、７…変換テーブル、８ａ～８ｄ…手続き、９…共有メモリ、１０…負荷測定部、１１…プロセス監視部、１２ａ～１２ｄ…サーバ・プロセス群、１３ａ～１３ｄ…データベースバッファ。

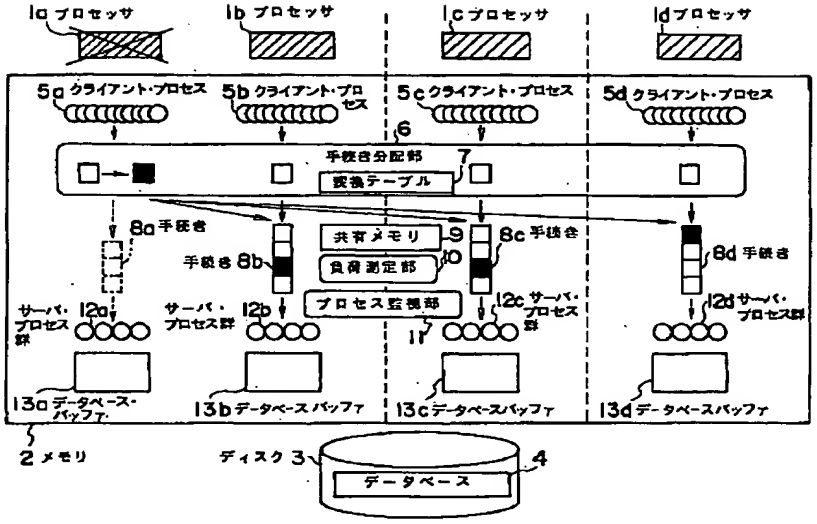
【図2】

変換前	変換後	割合
8a	8a	100%
8b	8b	100%
8c	8c	100%
8d	8d	100%

【図3】

変換前	変換後	割合
8a	8b	100%
8b	8b	100%
8c	8c	100%
8d	8d	100%

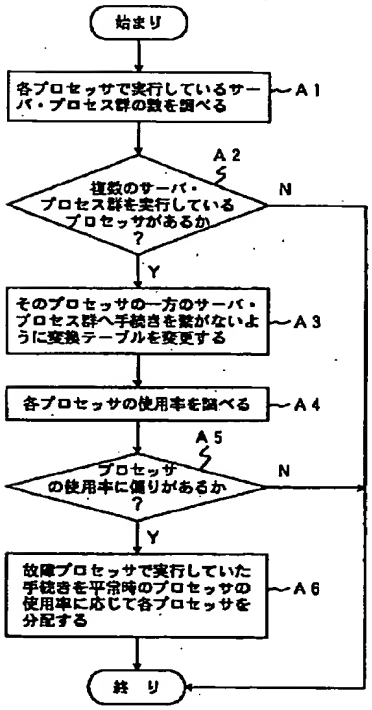
【 図1 】



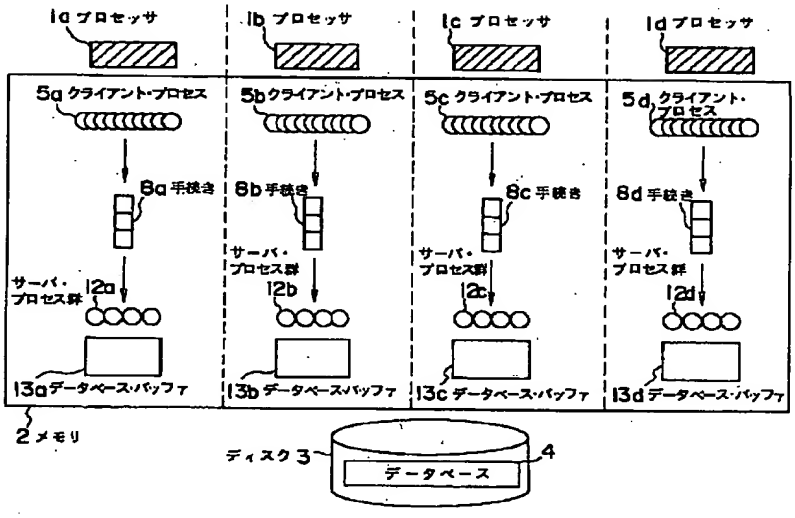
【 図4 】

変換前	変換後	割 合
8a	8b	33%
	8c	33%
	8d	33%
8b	8b	100%
8c	8c	100%
8d	8d	100%

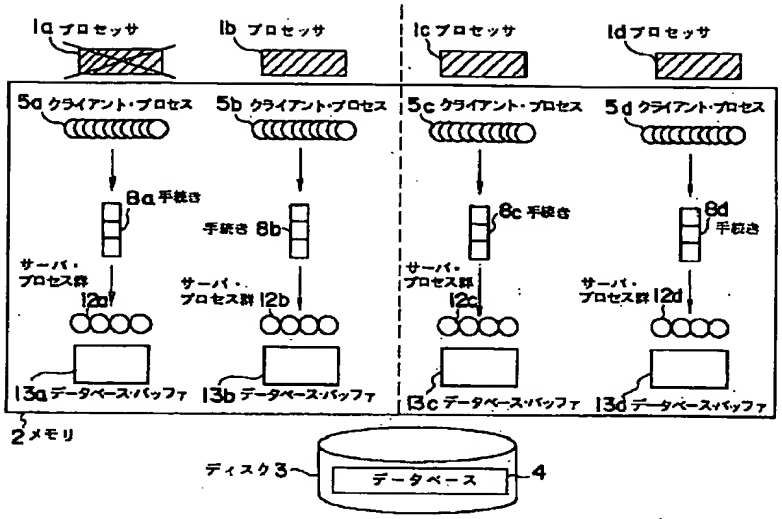
【 図5 】



【 図6 】



【 図7 】


1a プロセッサ
1b プロセッサ
1c プロセッサ
1d プロセッサ
5a クライアント・プロセス
5b クライアント・プロセス
5c クライアント・プロセス
5d クライアント・プロセス
8a 手続き
手続き 8b
8c 手続き
8d 手続き
サーバ・プロセス群 12a
サーバ・プロセス群 12b
サーバ・プロセス群 12c
サーバ・プロセス群 12d
13a データベース・バッファ
13b データベース・バッファ
13c データベース・バッファ
13d データベース・バッファ
2 メモリ
ディスク 3
データベース
4